SOFTWARE DEVELOPMENT TO ASSIST IN THE PROCESSING AND ANALYSIS OF DATA OBTAINED USING FIBER BRAGG GRATING INTERROGATION SYSTEMS

Intern: Rebecca Hicks

Advanced Structures and Measurements Group
Hugh L. Dryden Flight Research Center
Edwards, CA

Mentor: Allen Parker

8/3/2009

INTRODUCTION AND ABSTRACT

A fiber Bragg grating is a portion of a core of a fiber optic strand that has been treated to affect the way light travels through the strand. Light within a certain narrow range of wavelengths will be reflected along the fiber by the grating, while light outside that range will pass through the grating mostly undisturbed. Since the range of wavelengths that can penetrate the grating depends on the grating itself as well as temperature and mechanical strain, fiber Bragg gratings can be used as temperature and strain sensors. This capability, along with the light-weight nature of the fiber optic strands in which the gratings reside, make fiber optic sensors an ideal candidate for flight testing and monitoring in which temperature and wing strain are factors.

A team of NASA Dryden engineers has been working to advance the fiber optic sensor technology since the mid 1990's. The team has been able to improve the dependability and sample rate of fiber optic sensor systems, making them more suitable for real-time wing shape and strain monitoring and capable of rivaling traditional strain gauge sensors in accuracy. A fiber optic sensor system is capable of producing massive amounts of potentially useful data; however, methods to capture, record, and analyze all of this data in a way that makes the information useful to flight test engineers are currently limited. The purpose of this project is to research the availability of software capable of processing massive amounts of data in both real-time and post-flight settings, and to produce software segments that can be integrated to assist in the task as well.

According to the research, a combination of MATLAB and SigmaPlot seems to be the most appropriate selection for software to process the data, whereas IADS, with some

customized modifications, can be useful as well. Software segments created to assist with data collection include a User Datagram Protocol (UDP) broadcasting function module and a data compression function module.

The research conducted and function modules created will provide useful options for methods to capture, record, and analyze the data produced using fiber optic sensor systems on next-generation large aircrafts and on other future projects in which the sensors are used.

GOALS AND PURPOSE OF THE PROJECT

A fiber optic sensor system was recently tested on the Ikhana unmanned aircraft and will be used on future unmanned aircrafts. Since a fiber Bragg grating sensor can be placed every half-inch on each optic fiber, fibers of approximately 40 feet in length will have 960 sensors. Eight of these fibers (7,680 sensors total) can be simultaneously interrogated on each system with a maximum sample rate of about 60 Hz, meaning each system can produce 460,800 data points every second. Multiple systems may be used on a single aircraft, depending on how much of the area of the aircraft is to be monitored by the fiber optic sensor system. The amount of data produced on any given flight test can quickly become staggering.

The software selected to manage the data produced by a fiber optic sensor system must be able to: (1) process massive amounts of data at a speed useful in real-time settings (small fractions of a second); (2) process data in post-flight settings to allow test reproduction or further data analysis, inclusive; (3) produce, or make easier to produce, three-dimensional plots/graphs to make the data accessible to flight test engineers; and (4) be customized to allow users to use their own processing formulas or functions and display the data in formats they prefer. Several software programs were evaluated to determine their utility in completing the research objectives. These programs include: OriginLab, Graphis, 3D Grapher, Visualization Sciences

Group (VSG) Avizo Wind, Interactive Analysis and Display System (IADS), SigmaPlot, and MATLAB.

Of those evaluated, the top choices for a software program already in existence to be used are IADS, SigmaPlot, and MATLAB. Whereas IADS is the most customizable program and can handle the large data stream in real-time, the program has limited three-dimensional display capabilities. IADS is so customizable that a three-dimensional display can, theoretically, be designed, written in a programming code, and integrated into the software system, but the time and effort to make this happen may not be worthwhile when other processing software already exists. Whereas SigmaPlot has the best three-dimensional display capabilities and a plethora of other general data display options, the program has limited customizability and may not be able to handle the necessary amount of real-time processing. MATLAB is a diverse enough program to be able to meet all of the necessary qualifications, but installing MATLAB on every machine used for testing or data collection is not efficient or cost-effective. Either a standalone MATLAB executable or a mixture of MATLAB and SigmaPlot capabilities (a MATLAB executable that can be customized to receive and process data, then pass that data to SigmaPlot for display) may be the best option at this point.

Useful software segments include, but are not limited to: (1) a function module that can efficiently broadcast segments of data over a network; and (2) a function module that can compress and decompress data to increase the efficiency of onboard data storage, and therefore increase the flight time and range of which the fiber optic sensor systems are capable.

A small function module that uses User Datagram Protocol (UDP) to broadcast data over the network has been created. A UDP method allows the transmission of data in a way that takes less time than an equivalent Transmission Control Protocol (TCP) method. The function module

separates the data into uniform segments and then sends those segments over the network with other relevant information attached (identification of the original complete set of data, the segment's position, etc.).

A function module that multiplexes between a set of four compression algorithms is currently being created. All of the algorithms were tested on a 36,628KB text file to obtain a comparable compression speed and ratio using the same processor. One of the algorithms, entitled LZRW1/KH and written by Kurt Haenen, is the fastest (2.78 seconds), but compresses the least (final file is 18,961KB, about 51.77% of the original size). The other three algorithms are versions of one comprehensive algorithm (entitled ZPAQ and written by Matt Mahoney) that has different configurations (fastest: 742.53 seconds, 11,255KB or 30.73%; middle speed: 219.09 seconds, 7,245KB or 19.78%; slowest: 13,828.94 seconds, 7,091KB or 19.36%). Portions of the algorithms were rewritten or omitted to make the set of them work together within one program. A portion of the ZPAQ programming code was rewritten to make the three configuration files the original version needs to run unnecessary. The function module allows the user to have options for compressing data real-time. For instance, if a flight test is to use a higher sampling rate, a faster compression rate will be necessary. If the test is to be lengthy, a higher compression ratio may be more preferable.

IMPACT OF THE MUST INTERNSHIP ON MY CAREER GOALS

The MUST Internship has had a major impact on both my professional goals and my personal goals. My main goal remains the same: I want to have a position working in the field of computer science in which I feel competent and yet challenged in my daily tasks. However, the MUST internship has exposed me to career options I had not considered, and in doing so, has expanded my list of desired career choices. The internship has given me the opportunity to

experience the daily life of someone working in the computer science field at a NASA facility. Throughout my participation in the MUST program, I have also been exposed to information about many NASA education programs for which I can apply in the future that can help me obtain academic and professional goals. I have always intended to pursue a master's degree after completing my undergraduate study, but after learning of the multitude of opportunities available through NASA alone, I may pursue a master's degree sooner than I had planned.

My goal used to be to become an application software programmer at a software-based company and eventually work my way up to being a software tester or a software engineer. However, there are many opportunities available to those in technology fields. My experience in the MUST internship has opened my eyes to computer science support of flight testing and the possibility of writing or working with software for aerospace and data analysis systems. Working at Dryden Flight Research Center and visiting the Jet Propulsion Laboratory and Kennedy Space Center has shown me that NASA can offer a more exciting, dynamic, and flexible environment than a general private software-based company.

Working at Dryden Flight Research center has allowed me to see the difference between what tools and knowledge are used in an academic setting versus what tools and knowledge are necessary in the industry. The software tools I have learned about and used in classes have been useful, but I have also learned of software tools that are not usually covered at my school. Learning about this gap in my knowledge base has given me a chance to close it with narrowed, extra study and practice outside of class during my final year of undergraduate study.